

Compte rendu Javascript

Sommaire

Introduction	1
Fondamentaux 1	2
Fondamentaux 2	2
Fondamentaux 3	11
Conclusion	17

Introduction

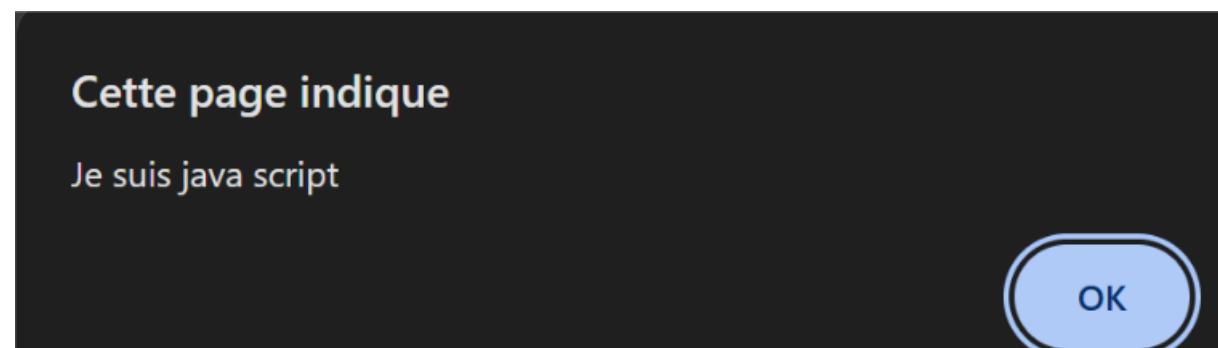
Ce compte rendu de TP en JavaScript vise à retracer notre découverte des bases du langage. Nous avons appris à manipuler des variables, afficher des informations via des alertes, utiliser des structures de contrôle comme *if/else* et *switch*, comprendre les opérateurs logiques, et mettre en place différentes boucles (*for*, *while*). L'objectif était surtout d'intégrer la logique fondamentale du langage et de voir comment ces outils interagissent entre eux.

Fondamentaux 1

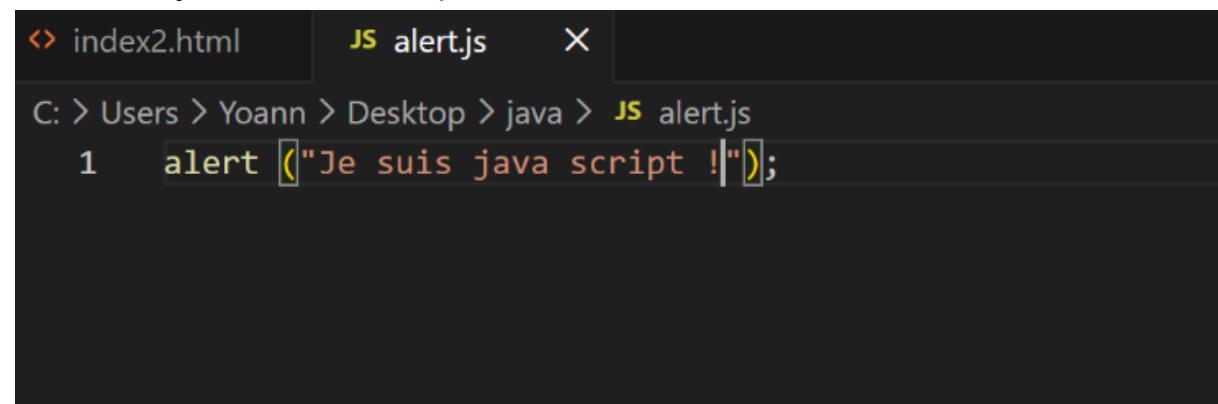
Premièrement je commence par afficher une alerte

```
<script>
    |     alert("Je suis java script ");
</script>
```

Résultat de l'alerte



Maintenant je fait exactement pareil mais dans un fichier externe



Fondamentaux 2

J'ai d'abord déclaré deux variables : une admin et une autre name.

J'ai assigné la valeur "John" à name, puis ai copié la valeur de name à admin.

Puis J'affiche la valeur d'admin en utilisant alert

```
1 let admin;  
2 let name;  
3  
4 name = "John";  
5  
6 admin = name;|  
7  
8 alert(admin);
```

Ce code a pour résultat

Cette page indique

John

OK

j'ai créé la variable avec le nom de la planète, puis aussi créer la variable pour stocker le nom du visiteur actuel :

```
const NOTRE_PLANETE = "La terre";  
let visiteur;
```

```
1 let name = "Ilya";
2
3 alert(`hello ${1}`);
4
5 alert(`hello ${"name"}`);
6
7 alert(`hello ${name}`);
```

La sortie du script sera :

hello 1
hello name
hello Ilya

Pour demander un nom et l'afficher

```
let name;
name = prompt("Saisir votre nom:");
alert(name)
```

Quelles sont les valeurs finales de toutes les variables **a**, **b**, **c** et **d** après le code ci-dessous ?

```
1 let a = 1, b = 1;
2
3 let c = ++a; // ?
4 let d = b++; // ?
```

Réponse : La valeur de A = 1 , B = 1 ,C = 2 et D = 1

Résultat d'affectation

Quelles sont les valeurs de **a** et **x** après le code ci-dessous ?

```
1 let a = 2;
2
3 let x = 1 + (a *= 2);
```

Valeur de A = 4
Valeur de X = 5

Quels sont les résultats de ces expressions ?

```
1  "" + 1 + 0
2  "" - 1 + 0
3  true + false
4  6 / "3"
5  "2" * "3"
6  4 + 5 + "px"
7  "$" + 4 + 5
8  "4" - 2
9  "4px" - 2
10 " -9 " + 5
11 " -9 " - 5
12 null + 1
13 undefined + 1
14 " \t \n" - 2
```

```
"" + 1 + 0;           // "10"
"" - 1 + 0;           // -1
true + false; // 1
6 / "3";           // 2
"2" * "3"; // 6
4 + 5 + "px"; // "9px"
"$" + 4 + 5; // "$45"
"4" - 2;           // 2
"4px" - 2;          // NaN
" -9 " + 5;          // "-9 5"
null + 1;           // 1
undefined + 1; // NaN
" \t \n" - 2; // -2
```

Voici un code qui demande à l'utilisateur deux nombres et affiche leur somme.

Cela ne fonctionne pas correctement. La sortie dans l'exemple ci-dessous est `12` (pour les valeurs d'invite par défaut).

Pourquoi ? Réparez-le. Le résultat doit être `3`.

```
1 let a = prompt("First number?", 1);
2 let b = prompt("Second number?", 2);
3
4 alert(a + b); // 12
```

```
let a = prompt ("Enter first number:", 1);
let b = prompt ("Enter second number:", 2);
```

```
alert [+ a + + b]; // 12
```

Quel sera le résultat pour les expressions suivantes :

```
1 5 > 4
2 "apple" > "pineapple"
3 "2" > "12"
4 undefined == null
5 undefined === null
6 null == "\n0\n"
7 null === +"\n0\n"
```

```
5 > 4 // true
"apple" > "pineapple" // false
"2" > "12" // true
undefined == null // true
undefined === null // false
null == "\n0\n" // false
null === +"\n0\n" // false
```

Si on met if une chaîne de caractère avec 0 alors l'alert s'affichera

Est-ce que `alert` sera affiché ?

```
1 if ("0") {
2   alert('Hello');
3 }
```

Ecrivez le code qui demande : ‘Quel est le nom “officiel” de JavaScript?’ :

```
let demande = prompt('Quel est le nom "Officiel" de JavaScript?')

if (demande == "ECMAScript") {
  alert('Bonne réponse');
} else [
  alert('Ne sait pas ? ECMAScript !');
}
```

Afficher le signe

En utilisant `if..else`, écrivez le code qui obtient un numéro via le `prompt`, puis l'affiche en `alert` :

- `1`, si la valeur est supérieure à zéro,
- `-1`, si inférieur à zéro,
- `0`, si est égal à zéro.

```
let demande = prompt('Donner un nombre :');

if (demande > 0) {
    alert('1');
} else if (demande < 0) {
    alert('-1');
} else {
    alert('0');
}
```

Réécrivez ce `if` en utilisant l'opérateur conditionnel `'?'` :

```
1 let result;
2
3 if (a + b < 4) {
4     result = 'Below';
5 } else {
6     result = 'Over';
7 }
```

```
let resultat;
```

```
(a + b < 4) ?
    alert('below') : alert('over');
```

Réécrire ce `if..else` en utilisant plusieurs opérateurs ternaires `'?'`.

Pour plus de lisibilité, il est recommandé de diviser le code en plusieurs lignes.

```
1 let message;
2
3 if (login == 'Employee') {
4   message = 'Hello';
5 } else if (login == 'Director') {
6   message = 'Greetings';
7 } else if (login == '') {
8   message = 'No login';
9 } else {
10   message = '';
11 }
```

```
let message;

(login == 'Employee') ? message = 'Hello' :
(login == 'Director') ? message = 'Greetings' :
(login == '') ? message = 'No login' :
message = '';
```

Quel est le résultat de OR ?

Qu'est-ce que le code ci-dessous va sortir ?

```
1 alert( null || 2 || undefined );
```

Résultat affiché : alerte affiche 2

Quel est le résultat des alertes OR ?

Qu'est-ce que le code ci-dessous va sortir ?

```
1 alert( alert(1) || 2 || alert(3) );
```

Les alertes affichées sont : 1, puis 2.
3 n'est jamais affiché car l'évaluation s'arrête à 2

Quel est le résultat de AND ?

Qu'est-ce que ce code va afficher ?

```
1 alert( 1 && null && 2 );
```

Le résultat affiché est : alert affiche null. (2 n'est jamais atteint)

Quel est le résultat des alertes AND ?

Qu'est-ce que ce code va afficher ?

```
1 alert( alert(1) && alert(2) );
```

Les alertes qui sont affichées sont : 1, puis undefined

Le résultat de OR AND OR

Quel sera le résultat ?

```
1 alert( null || 2 && 3 || 4 );
```

Le résultat affiche : alerte affiche 3

Vérifiez la plage entre

Ecrivez une condition "if" pour vérifier que l'age est compris entre 14 et 90 ans inclus.

"Inclus" signifie que l'age peut atteindre les 14 ou 90 ans.

```
let age = 25;  
  
✓ if (age >= 14 && age <= 90) {  
    alert("Âge compris entre 14 et 90.");  
}
```

Vérifiez à l'extérieur de la plage

Ecrivez une condition if pour vérifier que l'age n'est PAS compris entre 14 et 90 ans inclus.

Créez deux variantes: la première utilisant NOT !, La seconde – sans ce dernier.

```
let age = 100;  
  
if (!(age >= 14 && age <= 90)) {  
    alert("Âge NON compris entre 14 et 90 (Utilisation de NOT).");  
}
```

Une question à propos de "if"

Lesquelles de ces alertes vont s'exécuter ?

Quels seront les résultats des expressions à l'intérieur de if (...) ?

```
1 if (-1 || 0) alert( 'first' );  
2 if (-1 && 0) alert( 'second' );  
3 if (null || -1 && 1) alert( 'third' );
```

Les alertes qui vont s'exécuter sont alert('first') et alert('third')

Check the login

Écrivez le code qui demande une connexion avec `prompt`.

Si le visiteur entre "Admin", puis `prompt` pour un mot de passe, si l'entrée est une ligne vide ou Esc – affichez "Canceled", s'il s'agit d'une autre chaîne de caractères – alors affichez "I don't know you".

Le mot de passe est vérifié comme suit :

- S'il est égal à "TheMaster", alors affichez "Welcome!",
- Une autre chaîne de caractères – affichez "Wrong password",
- Pour une chaîne de caractères vide ou une entrée annulée, affichez "Canceled".

```
let userName = prompt("Qui êtes-vous ?", "");
if (userName === 'Admin') {
    let password = prompt("Mot de passe ?", "");
    if (password === 'TheMaster') {
        alert('Welcome!');
    } else if (password === '' || password === null) {
        alert('Canceled');
    } else {
        alert('Wrong password');
    }
} else if (userName === '' || userName === null) {
    alert('Canceled');
} else {
    alert("I don't know you");
```

Fondamentaux 3

Dernière valeur de boucle

Quelle est la dernière valeur affichée par ce code ? Pourquoi ?

```
1 let i = 3;
2
3 while (i) {
4     alert( i-- );
5 }
```

Dernière valeur affichée : 1.

Pourquoi :

Départ : $i = 3$.

Itération 1 : `while(3)` est vrai. `alert(3)` est exécuté. $i--$ décrémente i à 2 après l'alerte.

Itération 2 : `while(2)` est vrai. `alert(2)` est exécuté. $i--$ décrémente i à 1 après l'alerte.

Itération 3 : `while(1)` est vrai. `alert(1)` est exécuté. $i--$ décrémente i à 0 après l'alerte.

Itération 4 : `while(0)` est faux (car 0 est la valeur falsy en JavaScript). La boucle s'arrête.

Quelles valeurs affiche la boucle while ?

A votre avis, quelles sont les valeurs affichées pour chaque boucle ? Notez-les puis comparer avec la réponse.

Les deux boucles affichent-elles les mêmes valeurs dans l'`alert` ou pas ?

1.

Le préfixe sous forme `++i` :

```
1 let i = 0;
2 while (++i < 5) alert( i );
```

2.

Le postfixe sous forme `i++` :

```
1 let i = 0;
2 while (i++ < 5) alert( i );
```

Préfixe sous forme : 1, 2, 3, 4

```
let n = 10;
for (let i = 2; i <= n; i++) {
    let estPremier = true;

    for (let j = 2; j * j <= i; j++) {
        if (i % j === 0) {
            estPremier = false;
            break;
        }
    }
    if (estPremier) {
        alert(i);
    }
}
```

```
1 ↴ if (browser == 'Edge') {
2   | alert("You've got the Edge!");
3 } else if (browser == 'Chrome'
4   | | browser == 'Firefox'
5   | | browser == 'Safari'
6 ↴   | | browser == 'Opera') {
7   | alert('Okay we support these browsers too');
8 ↴ } else {
9   | alert('We hope that this page looks ok!');
10 }
```

```
1 let a = +prompt('a?', '');
2
3 switch (a) {
4     case 0:
5         alert(0);
6         break;
7
8     case 1:
9         alert(1);
10    break;
11
12    case 2:
13    case 3:
14        alert('2,3');
15        break;
16 }
```

```
function pow(x, n) {
    let result = x;

    for (let i = 1; i < n; i++) {
        result *= x;
    }

    return result;
}

let x = prompt("x?", '');
let n = prompt("n?", '');

if (n < 1) {
    alert(`Power ${n} is not supported, use a positive integer`);
} else {
    alert( pow(x, n) );
}
```

```
function min(a, b) {
  if (a < b) {
    return a;
  } else {
    return b;
  }
}

function min(a, b) [
  return a < b ? a : b;
]

function checkAge(age) {
  return (age > 18) ? true : confirm('Did parents allow you?');
}

function checkAge(age) {
  return (age > 18) || confirm('Did parents allow you?');
}

let a = +prompt('a?', '');

switch (a) {
  case 0:
    alert(0);
    break;

  case 1:
    alert(1);
    break;

  case 2:
  case 3:
    alert('2,3');
    break;
}
```

Conclusion

Ce TP nous a permis d'acquérir les notions essentielles de JavaScript : déclarer et modifier des variables, gérer des conditions et répéter des actions grâce aux boucles. Cette mise en pratique concrète des bases théoriques nous offre un socle solide pour aller vers des développements web plus dynamiques et complets.